CLAIMS

What is claimed:

1. A method comprising:

receiving a frame of data transmitted across a network into a receiving buffer of a target computer system;

recognizing and initiating a copy of said frame of data, said initiating comprising identifying at least one available Application Memory (AM) buffer in which to store a payload of said frame of data;

storing a header of said frame of data to a Receiving Frame Descriptor (RFD), said header comprising at least one sequence number;

zero-copying said payload of said received frame of data to said at least one AM buffer.

2. The method of Claim 1 further comprising allocating a new AM buffer if at least one available AM buffer is not identified, said allocating comprising creating a Receiving Buffer Descriptor (RBD) within an available RBD queue entry, said RBD comprising a value corresponding to a start address and a value corresponding to a size of said new AM buffer.

3. The method of Claim 2 wherein said zero-copying is aborted if said at least available RBD queue entry is not identified.

4. The method of Claim 3 further comprising creating a flow descriptor if said new AM buffer is allocated, said flow descriptor identifying a zero-copy flow.

5. The method of Claim 2 wherein said identifying comprises searching a pre-defined number of RBD's within said RBD queue for an RBD corresponding to said at least one available AM buffer.

6. The method of Claim 5 wherein said search is based on a difference between a an AM buffer size and a size of said payload of said frame of data.

7. The method of Claim 6 wherein said search is aborted if said search is unsuccessful after searching said predefined number of RBD's within said RBD queue.

8. The method of Claim 7 wherein said RBD queue is indexed by a head pointer, a tail pointer and a Last Completed RBD Number (LCRN) pointer, said head pointer comprising a value corresponding to an address of a next available RBD entry within said RBD queue and said tail pointer comprising a value corresponding to an address of an oldest incomplete RBD within said RBD queue and said LCRN pointer comprising a value corresponding to a full AM buffer.

9. The method of Claim 8 wherein an RBD entry within said RBD queue corresponding to a full AM buffer is invalidated.

10. The method of Claim 9 wherein said invalidated RBD entry is used to store future or pending RBD's.

11. A machine-readable medium having stored thereon a set of instructions, said set of instructions, when executed by a processor, cause said processor to perform a method comprising:

allocating at least one Application Memory (AM) buffer in which to store a payload of at least one frame of data;

storing a payload of said at least one frame of data into said at least one AM buffer;

storing a header associated with said at least one frame of data into at least one Receiving Frame Descriptor (RFD);

dynamically allocating or de-allocating within a queue of Receiving Buffer Descriptors (RBD's) at least one RBD corresponding to said at least one allocated AM buffer.

12. The machine-readable medium of Claim 11 wherein said dynamically de-allocating said RBD comprises invalidating said at least one RBD within said queue after said at least one AM buffer is full.

13. The machine-readable medium of Claim 12 wherein said dynamically allocating said RBD comprises storing said at least one RBD to said queue after allocating said at least one AM buffer.

14. The machine-readable medium of Claim 13 wherein said at least one RBD comprises a start address of a payload of said at least one frame of data stored within said at least one AM buffer.

15. The machine-readable medium of Claim 14 wherein said RBD queue comprises at least one RBD corresponding to at least one AM buffer, said at least one AM buffer comprising at least one data frame payload.

16. The machine-readable medium of Claim 11 further comprising notifying a device driver after said at least one AM buffer is full.

17. The machine-readable medium of Claim 16 wherein a Last Completed RBD Number (LCRN) is stored within said RFD after said at least one AM buffer is full.

18. The machine-readable medium of Claim 17 wherein said notifying further comprises updating an LCRN pointer to point to an RBD within said RBD queue corresponding to said full at least one AM buffer.

19. The machine-readable medium of Claim 18 wherein said queue is indexed by a plurality of pointers, said pointers comprising a head, tail, and LCRN pointer.

20. The machine-readable medium of Claim 19 wherein said head pointer comprises a value corresponding to an address of a next available RBD entry within said queue and said tail pointer comprises a value corresponding to an address of an oldest incomplete RBD within said queue and said LCRN pointer comprises a value corresponding to an address of a last completed RBD within said queue.

21. A system comprising:

at least one CPU;

at least one buffer unit, said at least one buffer unit being coupled to said at least one CPU;

at least one memory unit to store at least one Application Memory (AM) buffer, said at least one memory unit being coupled to said at least one

CPU, said at least one AM buffer being associated with at least one entry within a Receiving Buffer Descriptor (RBD) queue, said RBD queue wherein at least one RBD is enabled to be dynamically allocated or de-allocated.

22. The system of Claim 21 wherein said at least one RBD is enabled to be dynamically de-allocated by invalidating said at least one RBD within said RBD queue after said at least one AM buffer is full, said at least one RBD being enabled to be dynamically allocated by adding said at least one RBD to said queue after said at least one AM buffer is allocated.

23. The system of Claim 22 wherein said at least one RBD comprises a value corresponding to a start address and a value corresponding to a size of said at least one AM buffer.

24. The system of Claim 23 wherein an indication is made to a software driver after said at least one AM buffer is full.

25. The system of Claim 24 wherein a Last Completed RBD Number (LCRN) is stored within a Receiving Frame Descriptor (RFD) after said at least one AM buffer is full, said RFD being associated with said at least one frame of data.

26. The system of Claim 25 wherein said indicating comprises updating an LCRN pointer to point to an RBD within said RBD queue corresponding to said full at least one AM buffer.

27. The system of Claim 26 wherein said queue is indexed by a plurality of pointers, said pointers comprising a head, tail, and LCRN pointer.

28. The system medium of Claim 27 wherein said head pointer comprises a value corresponding to an address of a next available RBD entry within said queue and said tail pointer comprises a value corresponding to an address of an oldest incomplete RBD within said queue and said LCRN pointer comprises a value corresponding to an address of a last completed RBD within said queue.

29. The system of Claim 28 wherein said RFD comprises a Zero Copy (ZC) bit and a flow identification (ID), said ZC bit being enable to indicate whether said at least one frame of data is associated with a ZC flow, said ZC flow being identified by said flow ID.

30. The system of Claim 29 wherein, if said ZC bit is set, said device driver de-allocates from said RBD queue an RBD referenced by said LCRN pointer, calculates a remaining number of valid RBD's within said RBD queue, and allocates a new RBD's to replace said de-allocated RBD.